

TASK- 3



# DIABETES PREDICTION ANALYSIS

Retrieve the Patient\_id and ages of all patients.

ANS-

Query

Query History

1

2

3

SELECT

patient\_id,

age

from

diabetes\_prediction

Data Output

Messages

Notifications

patient\_id

character varying (100)

age

character varying (10)

1

PT101

80

2

PT102

54

3

PT103

28

4

PT104

36

5

PT105

76

Total rows: 1000 of 100000

Query complete 00:00:00.108

**ANS-**

Query
Query History

```

1 SELECT * FROM diabetes_prediction
2 WHERE gender = 'Female' AND age > '40'

```

Data Output
Messages
Notifications

	employee_name	patient_id	gender	age	hypertension	heart_disease	smoking_history	bmi	hba1c_level	blood_glucose	diabetes
	character varying (100)	character varying (10)	character varying (10)	character varying (10)	boolean	boolean	character varying (10)	double precision	double precision	double precision	boolean
1	NATHANIEL FORD	PT101	Female	80	false	true	never	25.19	6.6	140	false
2	GARY JIMENEZ	PT102	Female	54	false	false	No info	27.32	6.6	80	false
3	ALSON LEE	PT107	Female	44	false	false	never	19.31	6.5	200	true
4	DAVID KUSHNER	PT108	Female	79	false	false	No info	23.86	5.7	85	false
5	ARTHUR KENNEY	PT111	Female	53	false	false	never	27.32	6.1	85	false
6	PATRICIA JACKSON	PT112	Female	54	false	false	former	54.7	6	100	false
7	EDWARD HARRIS	PT113	Female	78	false	false	former	36.05	5	130	false
8	JOHN MARTIN	PT114	Female	67	false	false	never	25.69	5.8	200	false
9	DAVID FRANKLIN	PT115	Female	76	false	false	No info	27.32	5	160	false
10	SEBASTIAN WOOD	PT118	Female	42	false	false	never	24.48	5.7	158	false
11	MARTY ROSS	PT119	Female	42	false	false	No info	27.32	5.7	80	false
12	GEORGE GARCIA	PT123	Female	69	false	false	never	21.24	4.8	85	false

Total rows: 33311 of 33311
Query complete 00:00:00.321
Ln 2, Col 39

Calculate the average BMI of patients.

ANS-

The screenshot shows a SQL query editor with the following components:

- Query Editor:** Contains the SQL query:

```
1 SELECT AVG(bmi) FROM diabetes_prediction
2
```
- Data Output Panel:** Displays the result of the query. A tooltip for the 'avg' function is visible, showing 'double precision' and a lock icon. The result table has one row:

	avg
1	27.32076709999422
- Status Bar:** At the bottom, it shows 'Total rows: 1 of 1', 'Query complete 00:00:00.108', and 'Ln 1, Col 8'.

List patients in descending order of blood glucose levels.

ANS-

QueryQuery History

1SELECT employeeName, patient\_id, blood\_glucose\_level FROM diabetes\_prediction

2ORDER BY blood\_glucose\_level DESC

3

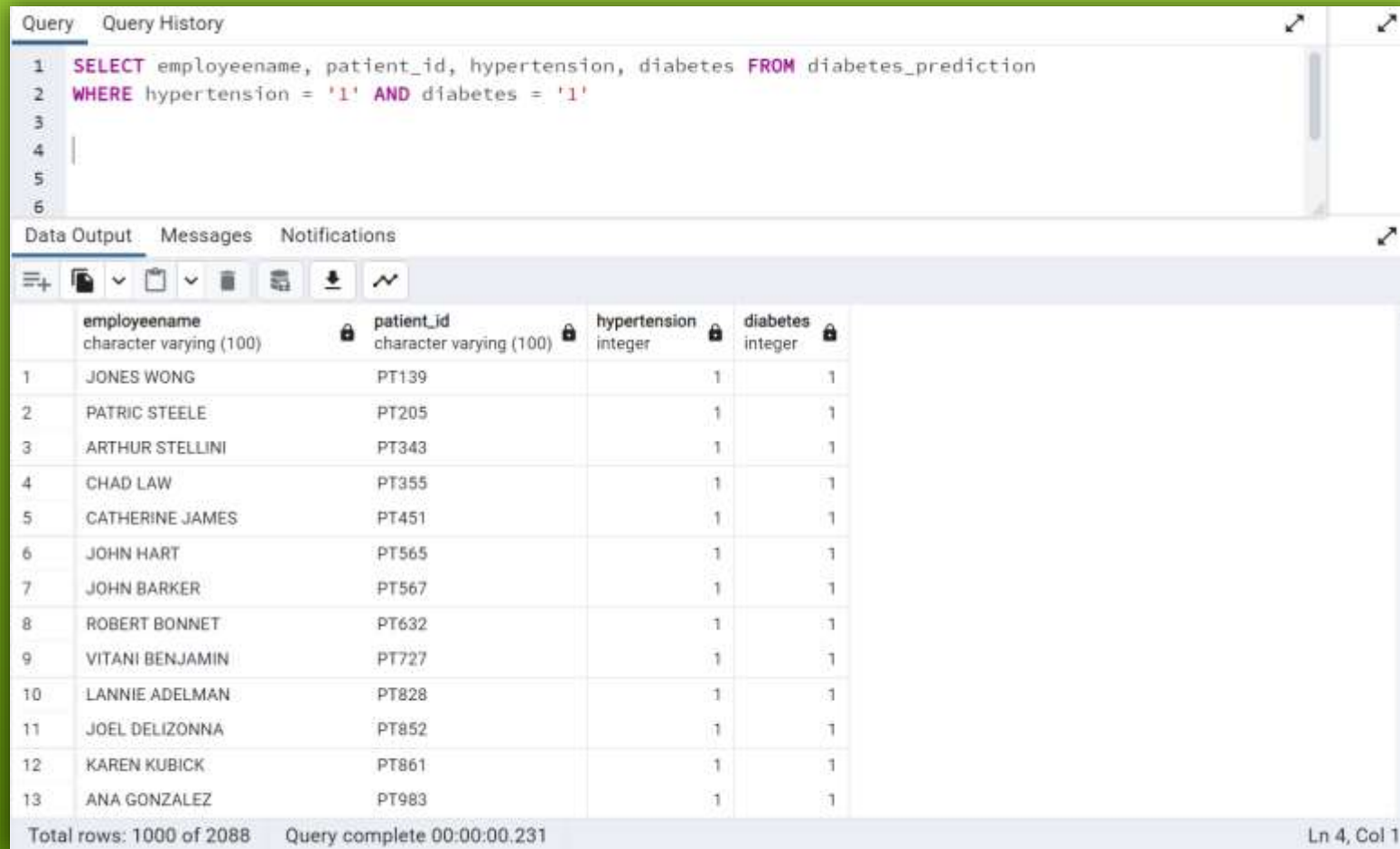
Data OutputMessagesNotifications

	employeeName character varying (100)	patient_id character varying (100)	blood_glucose_level double precision
1	MAUNG SEIN HO	PT20947	300
2	JOHN SCOTT	PT5302	300
3	SHERMAN YU	PT6379	300
4	CINDY TOM	PT16618	300
5	MARY JEAN TANYAO	PT22390	300
6	ANDREW MOLINA	PT16657	300
7	Michael Browne	PT36433	300
8	Anabella Alfaro	PT47863	300
9	RON BASCONCILLO	PT28544	300
10	PETER FONG	PT28834	300
11	Jane Ma	PT47871	300
12	JOSEPH HEID	PT32704	300
13	KIMBERLEY DAVIS	PT25412	300

Total rows: 1000 of 100000Query complete 00:00:00.259Ln 3, Col 1

Find patients who have hypertension and diabetes.

ANS-



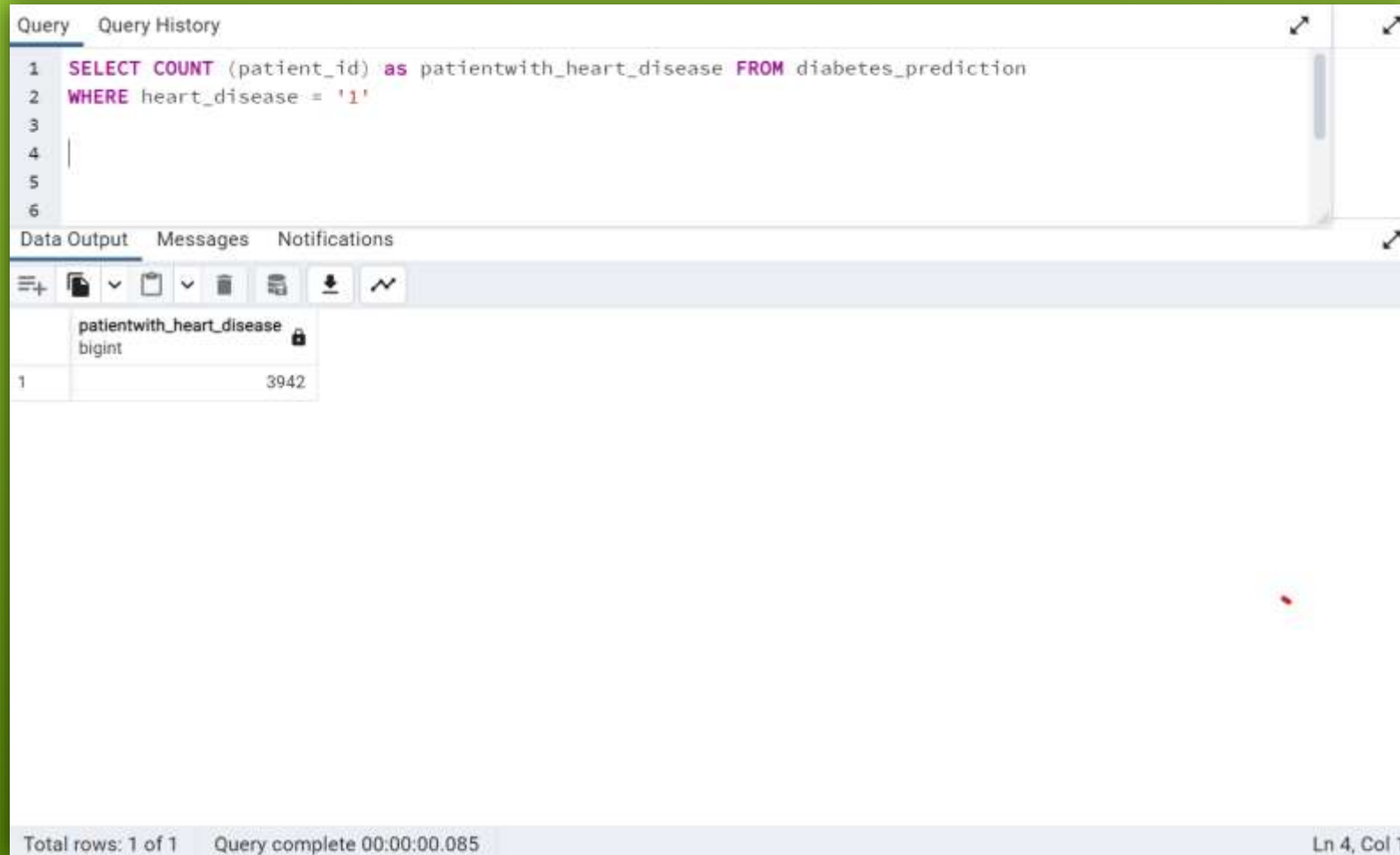
The screenshot shows a database query tool interface. The top pane displays a SQL query: `SELECT employeename, patient_id, hypertension, diabetes FROM diabetes_prediction WHERE hypertension = '1' AND diabetes = '1'`. The bottom pane shows the results of the query as a table with 13 rows. The columns are `employeename`, `patient_id`, `hypertension`, and `diabetes`. The status bar at the bottom indicates 'Total rows: 1000 of 2088' and 'Query complete 00:00:00.231'.

	employeename character varying (100)	patient_id character varying (100)	hypertension integer	diabetes integer
1	JONES WONG	PT139	1	1
2	PATRIC STEELE	PT205	1	1
3	ARTHUR STELLINI	PT343	1	1
4	CHAD LAW	PT355	1	1
5	CATHERINE JAMES	PT451	1	1
6	JOHN HART	PT565	1	1
7	JOHN BARKER	PT567	1	1
8	ROBERT BONNET	PT632	1	1
9	VITANI BENJAMIN	PT727	1	1
10	LANNIE ADELMAN	PT828	1	1
11	JOEL DELIZONNA	PT852	1	1
12	KAREN KUBICK	PT861	1	1
13	ANA GONZALEZ	PT983	1	1

Total rows: 1000 of 2088    Query complete 00:00:00.231    Ln 4, Col 1

Determine the number of patients with heart disease.

ANS-



The screenshot displays a SQL query execution window. The query is as follows:

```
1 SELECT COUNT (patient_id) as patientwith_heart_disease FROM diabetes_prediction
2 WHERE heart_disease = '1'
3
4
5
6
```

The results are shown in the Data Output tab, which contains a single row:

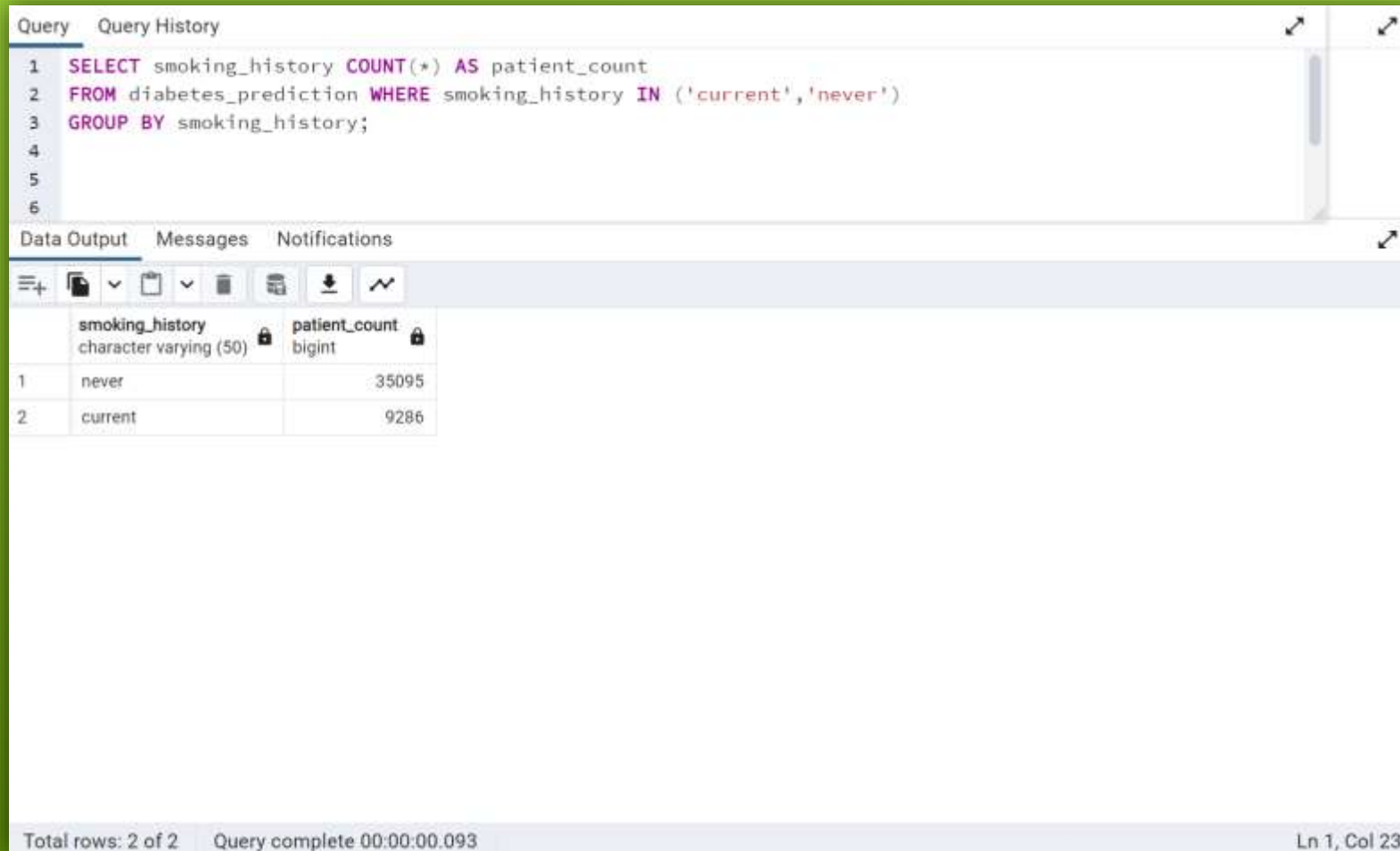
	patientwith_heart_disease
1	3942

The status bar at the bottom indicates "Total rows: 1 of 1" and "Query complete 00:00:00.085". The cursor is positioned at "Ln 4, Col 1".



Group patients by smoking history and count how many smokers and nonsmokers there are.

ANS-



The screenshot shows a SQL query editor with a query window and a data output window. The query is as follows:

```
1 SELECT smoking_history COUNT(*) AS patient_count
2 FROM diabetes_prediction WHERE smoking_history IN ('current','never')
3 GROUP BY smoking_history;
```

The data output window shows the results of the query:

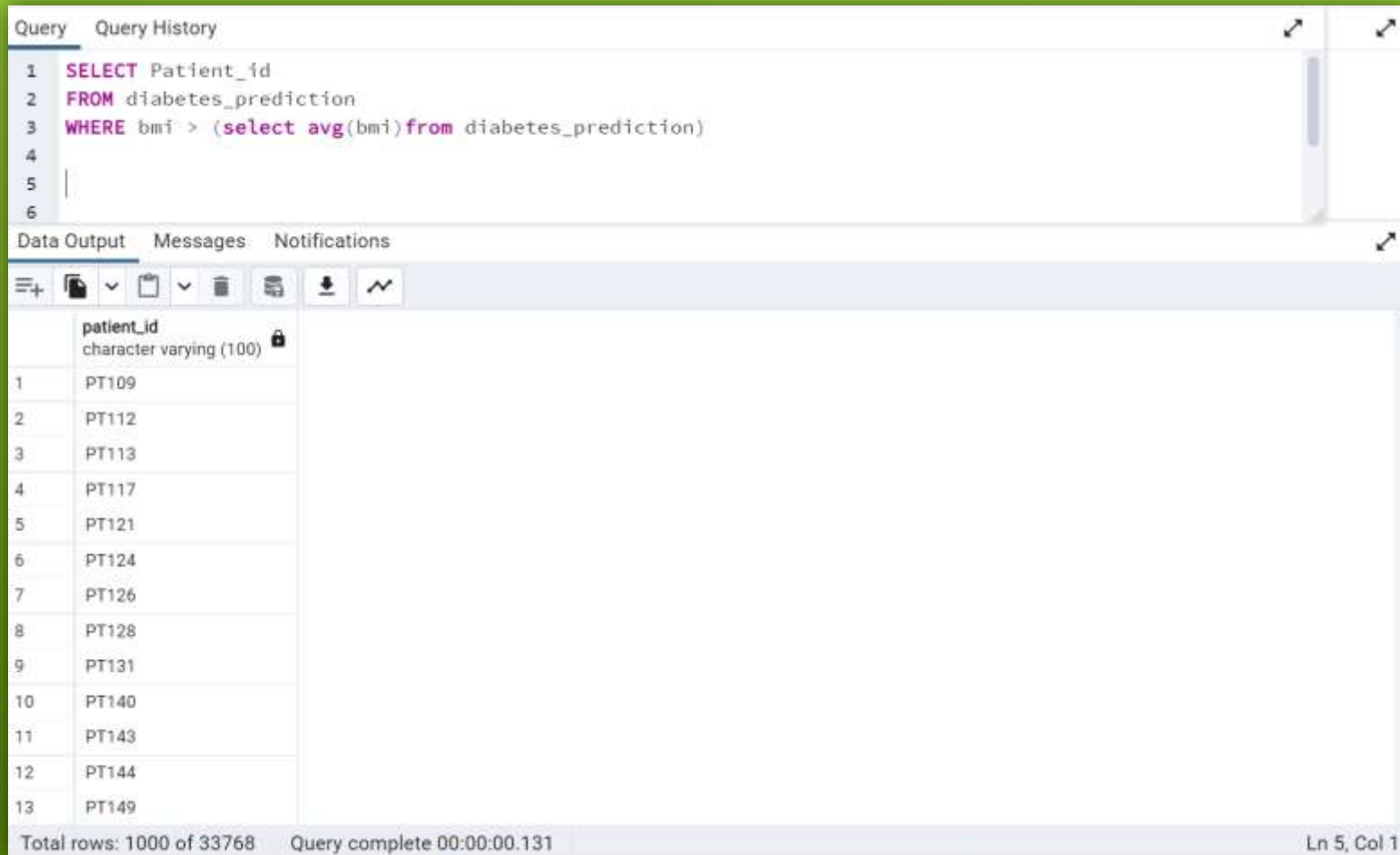
	smoking_history character varying (50)	patient_count bigint
1	never	35095
2	current	9286

Total rows: 2 of 2    Query complete 00:00:00.093    Ln 1, Col 23



Retrieve the Patient\_ids of patients who have a BMI greater than the average BMI.

ANS-



```
1 SELECT Patient_id
2 FROM diabetes_prediction
3 WHERE bmi > (select avg(bmi) from diabetes_prediction)
4
5
6
```

	patient_id character varying (100)
1	PT109
2	PT112
3	PT113
4	PT117
5	PT121
6	PT124
7	PT126
8	PT128
9	PT131
10	PT140
11	PT143
12	PT144
13	PT149

Total rows: 1000 of 33768    Query complete 00:00:00.131    Ln 5, Col 1

Find the patient with the highest HbA1c level and the patient with the lowest HbA1c level.

Query Query History

```
1 -- patient with the highest HbA1c level
2 SELECT patient_id, hba1c_level FROM diabetes_prediction
3 ORDER BY HbA1c_level DESC
4 LIMIT 1;
5
6
7 -- patient with the lowest HbA1c level
8 SELECT patient_id, hba1c_level FROM diabetes_prediction
9 ORDER BY HbA1c_level ASC
10 LIMIT 1;
```

Data Output Messages Notifications

	patient_id character varying (100)	hba1c_level double precision
1	PT141	9

Total rows: 1 of 1 Query complete 00:00:00.086

Query Query History

```
1 -- patient with the highest HbA1c level
2 SELECT patient_id, hba1c_level FROM diabetes_prediction
3 ORDER BY HbA1c_level DESC
4 LIMIT 1;
5
6
7 -- patient with the lowest HbA1c level
8 SELECT patient_id, hba1c_level FROM diabetes_prediction
9 ORDER BY HbA1c_level ASC
10 LIMIT 1;
```

Data Output Messages Notifications

	patient_id character varying (100)	hba1c_level double precision
1	PT120	3.5

Total rows: 1 of 1 Query complete 00:00:00.098

Rank patients by blood glucose level within each gender group.

ANS-

Query    Query History

```
1  SELECT patient_id,gender,blood_glucose_level,
2      RANK() OVER(PARTITION BY gender ORDER BY blood_glucose_level) AS Blood_glucose_levelRANK
3  FROM diabetes_prediction;
4
5
6
```

Data Output    Messages    Notifications

	patient_id character varying (100)	gender character varying (10)	blood_glucose_level double precision	blood_glucose_levelrank bigint
1	PT19978	Female	80	1
2	PT13302	Female	80	1
3	PT2580	Female	80	1
4	PT43598	Female	80	1
5	PT19069	Female	80	1
6	PT16796	Female	80	1
7	PT43594	Female	80	1
8	PT35160	Female	80	1
9	PT46495	Female	80	1

Total rows: 1000 of 100000    Query complete 00:00:00.376

Ln 5, Col 1

Update the smoking history of patients who are older than 50 to "Ex-smoker."

ANS-

Query

Query History

1

UPDATE diabetes\_prediction

2

SET smoking\_history = 'Ex-smoker'

3

WHERE age > 50

4

5

SELECT \* FROM diabetes\_prediction

6

WHERE age > 50

Data Output

Messages

Notifications

	employeename character varying (100)	patient_id character varying (100)	gender character varying (10)	age numeric	hypertension integer	heart_disease integer	smoking_history character varying (50)	bmi double precision	hba1c_level double precision
1	ERIC SOTO	PT5013	Male	52	1	0	Ex-smoker	25.79	
2	KYLE CURRY	PT7951	Male	54	0	0	Ex-smoker	27.32	
3	KEVIN LEE	PT13270	Male	52	0	0	Ex-smoker	27.32	
4	ANNA LAM	PT13427	Male	62	0	0	Ex-smoker	22.92	
5	ALLEN LEE	PT13690	Male	56	0	0	Ex-smoker	38.3	
6	VILOM TAM	PT14483	Male	73	0	0	Ex-smoker	28.55	
7	KYIN YI	PT14579	Female	76	0	0	Ex-smoker	23.82	
8	DOVE YU	PT16232	Female	61	0	0	Ex-smoker	33.25	
9	AIQUN XU	PT33264	Male	80	0	0	Ex-smoker	27.32	

Total rows: 34821 of 34821

Query complete 00:00:00.105

Ln 5, Col 1

Insert a new patient into the database with sample data.

ANS-

The screenshot shows a database query editor with the following SQL code:

```
1 INSERT INTO diabetes_prediction
2 VALUES('John Pollock', 'PT101', 'Male', 50, 0,0, 'Never', 25.22, 5.1, 151, 1)
3
4
5 SELECT * FROM diabetes_prediction
6 Where employeeename = 'John Pollock'
```

Below the query editor, the 'Data Output' tab is active, displaying a table with the following columns and data:

	employeeename character varying (100)	patient_id character varying (100)	gender character varying (10)	age numeric	hypertension integer	heart_disease integer	smoking_history character varying (50)	bmi double
1	John Pollock	PT101	Male	50	0	0	Never	

At the bottom of the window, the status bar indicates: 'Total rows: 1 of 1', 'Query complete 00:00:00.054', and 'Ln 5, Col 1'.

Delete all patients with heart disease from the database.

ANS-

QueryQuery History

1

2DELETE FROM diabetes\_prediction

3WHERE heart\_disease = 1

4

5

6SELECT \* FROM diabetes\_prediction

Data OutputMessagesNotifications

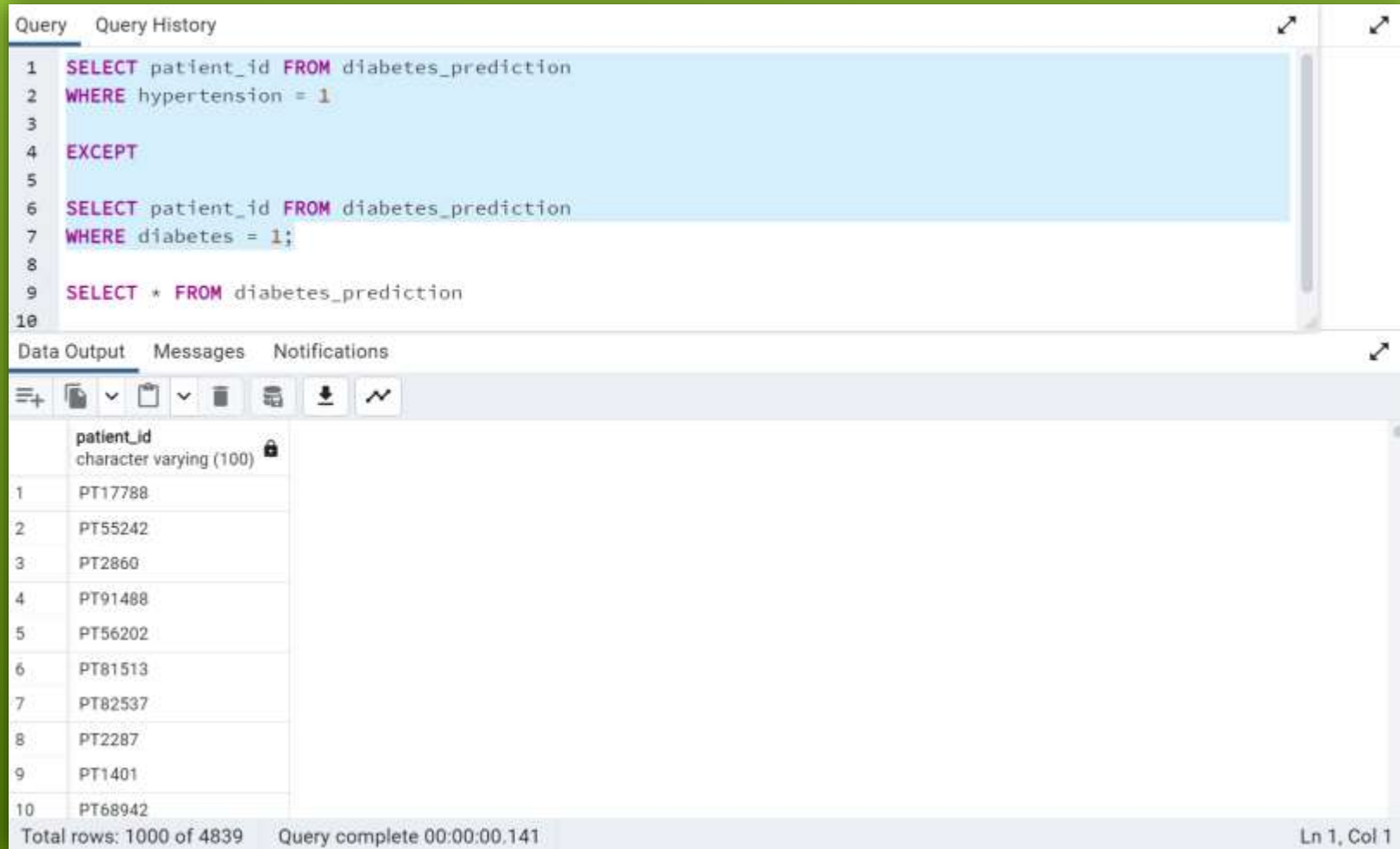
	employeename character varying (100)	patient_id character varying (100)	gender character varying (10)	age numeric	hypertension integer	heart_disease integer	smoking_history character varying (50)
1	John Pollock	PT102	Male	50	0	0	never
2	ALBERT PARDINI	PT103	Male	28	0	0	never
3	CHRISTOPHER CHONG	PT104	Female	36	0	0	current
4	DAVID SULLIVAN	PT106	Female	20	0	0	never
5	ALSON LEE	PT107	Female	44	0	0	never
6	MICHAEL MORRIS	PT109	Male	42	0	0	never
7	JOANNE HAYES-WHITE	PT110	Female	32	0	0	never
8	AMY HART	PT117	Male	15	0	0	never
9	SEBASTIAN WONG	PT118	Female	42	0	0	never
10	MARTY ROSS	PT119	Female	42	0	0	No info
11	ELLEN MOFFATT	PT120	Male	37	0	0	ever
12	VENUS AZAR	PT121	Male	40	0	0	current

Total rows: 1000 of 96059Query complete 00:00:00.305Ln 6, Col 1



Find patients who have hypertension but not diabetes using the EXCEPT operator

ANS-



The screenshot displays a SQL query editor with a query window and a data output window. The query window contains the following SQL code:

```
1 SELECT patient_id FROM diabetes_prediction
2 WHERE hypertension = 1
3
4 EXCEPT
5
6 SELECT patient_id FROM diabetes_prediction
7 WHERE diabetes = 1;
8
9 SELECT * FROM diabetes_prediction
10
```

The data output window shows the results of the query, which is a list of patient IDs. The first column is labeled 'patient\_id' and is of type 'character varying (100)'. The results are as follows:

patient_id
PT17788
PT55242
PT2860
PT91488
PT56202
PT81513
PT82537
PT2287
PT1401
PT68942

The status bar at the bottom indicates 'Total rows: 1000 of 4839', 'Query complete 00:00:00.141', and 'Ln 1, Col 1'.



Define a unique constraint on the "patient\_id" column to ensure its values are unique.

ANS-

QueryQuery History

1ALTER TABLE diabetes\_prediction

2ADD CONSTRAINT unique\_patient\_id

3UNIQUE (patient\_id)

4

5

6SELECT \* FROM diabetes\_prediction

7

8

9

10

Data OutputMessagesNotifications

	employeenam character varying (100)	patient_id character varying (100)	gender character varying (10)	age numeric	hypertension integer	heart_disease integer	smoking_history character varying (50)
1	ALBERT PARDINI	PT103	Male	28	0	0	never
2	CHRISTOPHER CHONG	PT104	Female	36	0	0	current
3	DAVID SULLIVAN	PT106	Female	20	0	0	never
4	ALSON LEE	PT107	Female	44	0	0	never
5	MICHAEL MORRIS	PT109	Male	42	0	0	never
6	JOANNE HAYES-WHITE	PT110	Female	32	0	0	never
7	AMY HART	PT117	Male	15	0	0	never
8	SEBASTIAN WONG	PT118	Female	42	0	0	never
9	MARTY ROSS	PT119	Female	42	0	0	No Info

Total rows: 1000 of 96058Query complete 00:00:00.334Ln 6, Col 1

Create a view that displays the Patient\_ids, ages, and BMI of patients.

ANS-

The screenshot shows a SQL IDE interface. The top pane displays the following SQL code:

```
1 CREATE VIEW patient_bmi_view AS
2 SELECT patient_id, age, bmi FROM diabetes_prediction;
3
4
5
6 SELECT * FROM patient_bmi_view
7
8
9
10
```

The bottom pane shows the results of the query, displaying a table with 10 rows and 3 columns: patient\_id, age, and bmi. The status bar at the bottom indicates 'Total rows: 1000 of 96058' and 'Query complete 00:00:00.152'.

	patient_id character varying (100)	age numeric	bmi double precision
1	PT103	28	27.32
2	PT104	36	23.45
3	PT106	20	27.32
4	PT107	44	19.31
5	PT109	42	33.64
6	PT110	32	27.32
7	PT117	15	30.36
8	PT118	42	24.48
9	PT119	42	27.32
10	PT120	37	25.72

Total rows: 1000 of 96058    Query complete 00:00:00.152    Ln 6, Col 1

# THANK YOU

Atharva Patil